ملاحظات هامة :

1. لا يشترط استيعاب كل نقطة في هذا النموذج في معيار النجاح هنا بل يعفى عن بعض الإغفالات والأخطاء ، فلو نسي مثلا بعض بنى أو تقنيات البرمجة ، أو بعض الأمثلة من النظام أو بعض مبادئ التصميم أو معايير الجودة فلا بأس ، ولكن يفضل أن ينبه المعلم الطالب إلى الأمور الأساسية .
2. يفضل عدم إدراج رموز المعايير في التقرير (مثل AP1,AP2 ...) ويمكن الاكتفاء بالعنوان .
3. لمن أراد ان يتقدم للتفوق والامتياز، لا يشترط فصل الفقرات إلى نجاح وتفوق وتميز بل يمكن دمجها معاً وهو الأفضل .
4. يوجد أيضا أمثلة أخرى في التطبيق على مهارات التفكير الحاسوبي أو مبادئ المنطق البرمجي أو غيرها ولكن ذكرت أمثلة فقط للتوضيح .
5. لن أطبق مهارات التفكير الحاسوبي على أكثر من نظام ، والمطلوب نظام واحد كما هو في المواصفات والذي يعتبر المرجع ، كما بُلغنا سابقاً .
6. يجب كتابة المراجع بطريقة صحيحة ، خاصة عند كتابة رابط موقع فيجب أن يكون فعال ويشير إلى المعلومات الصحيحة المتعلقة بالموضوع .
7. دائما في المقارنة يفضل عمل جداول كما في هذا النموذج.
8. يفضل عمل خرائط ورسومات لزيادة جودة التقرير .

طريقة إعداد تقرير لمهمة أ_ البرمجة

- طبعا قبل كل شي يدرج الطالب الترويسة والتي تشمل اسمه ورقمه ويفضل بالإنجليزي مع شعار أو كلمة BTEC.
- يكتب عنوان المهمة بخط كبير وعريض .
- يعمل فهرس .
- يبدأ بمقدمة عامة (ممكن أن يشير فيها إلى معلومات عن البرنامج النموذجي والمشاكل والاحتياجات المحددة فيه والمراد حلها ، لكنه ليس ضروري).

Naji Kamamji + Ashraf Ganem

AP1 : شرح كيف يتم تطبيق مهارات التفكير الحسابي في إيجاد الحلول التي يمكن تفسيرها إلى تطبيق

❖ **يشرح عن المهارات الأربعة بشكل عام** (ماذا تفعل هذه المهارة) مع التطبيق (على نظام المكتبة مثلا) :

أ- **التفكيك :** "هو تقسيم المشكلة إلى أجزاء أو مشكلات أصغر بحيث ..الخ ، كما في الكتاب " .

- **التطبيق على النظام :** من خلال فحص النظام نلاحظ أنه تم تقسيم المشكلة إلى مشكلات فرعية مثل:
  - إضافة كتاب جديد ، باستخدام دالة : `AddBook()`
  - استعارة الكتاب : `LoanBook()`
  - التنقل بين أجزاء النظام : الدالة `Main()`
  - ...الخ .

ب. **تعرُّف الأنماط :** " هو القدرة على تحديد السمات المشتركة ...الخ ، كما في الكتاب "

- **التطبيق على النظام :** نلاحظ وجود سمات وخطوات متكررة لكل عملية من العمليات التالية :
  - **نمط التحقق:** نلاحظ أن هناك نمطاً (`if (!books.ContainsKey(id))`) متكرراً يتكرر في دالتي: `LoanBook()` و `ReturnBook ()`
  - **نمط التكرار :** `Foreach` .....
  - **نمط هيكلية (بنية) الكتاب :** (`Class Book`) ....

ج. **تعميم النمط وتجريده :** هو تصفية المعلومات غير الضرورية والتركيز فقط على المعلومات المطلوبة لحل المشكلة.

- **التطبيق على النظام :** نلاحظ بدلاً من التعامل مع كل خاصية على حدة، تم تجميعها في كلاس واحد وأيضا أخفى التفاصيل الغير ضرورية مثل تاريخ الإصدار ، دار النشر ،....الخ ، كالتالي :

```
class Book
{
  public string Title;
  public int TotalCopies;
  public int AvailableCopies;
}
```

Najl Kamamji - Ashraf Ganem

مع ملاحظة انه يمكن تعميم هذا الكلاس لجميع الكتب أو تعميمه لفئات تشبهه مثل المجلات الأقراص ...الخ .

د- تمثيل عناصر المشكلة : هنا يتم تحديد الثوابت ،المتغيرات، المدخلات ، المخرجات، والعمليات وهياكل البيانات المناسبة لتمثيل الحل (كخوارزمية للحل) .

- **التطبيق على النظام :** نلاحظ وجود كل ما ذكر في البرنامج تقريبا كالتالي :
  - **الثوابت :** تم استخدام const string DATA_FILE لتمثيل اسم الملف، لأنه قيمة ثابتة لا تتغير أثناء تشغيل البرنامج.
  - **المتغيرات :** مثل ID و Total لتمثيل مدخلات المستخدم المتغيرة.
  - **هياكل البيانات :** مثل Stack لتمثيل آخر الكتب المسترجعة و ...الخ .
  - **العمليات** : مثل الدوال التي تمثل استعارة كتاب و....الخ .
  - ...الخ .

AP2 : شرح كيفية تطبيق مبادئ البرمجة الحاسوبية بلغات مختلفة لإنتاج تطبيقات برمجية.

- **يفضل أن يشرح عن البرامج وما يتعلق بها من مفاهيم أساسية (مثل : البرمجة ، لغات البرمجة وانواعها (عالية أو منخفضة المستوى)، المترجم ، المفسر ...الخ .**
- **يشرح عن نماذج البرمجة المتاحة مع ما يمثلها من لغات برمجة ، أو العكس (أي مجموعة لغات البرمجة المتاحة وإلى أي نموذج تنتمي (لكن سأعتمد طريقة الكتاب وعمل جدول لذلك) كالتالي :**

النموذج	اللغات التي تمثلها	ميزاتها وخصائصها
الإجرائي	**Basic ,C,Fortran**	هي طريقة في البرمجة تتم من خلال كتابة تعليمات - تتركز في الغالب داخل دوال أو إجراءات - على شكل خطوات محددة ومتسلسلة لحل مشكلة معينة . حيث تمتاز بـ .....
الكائنية		
الموجهة بالأحداث		
الآلة		
الترميز		
البرمجة النصية		

Najmeddine Ashraf Garem

❖ يقارن بين اللغات المختلفة (ويختار 3 حسب طلب الوزارة) من حيث العوامل المذكورة في الجدول :

ملاحظة: لا يشترط في مرحلة النجاح هنا أن تذكر السبب عند المقارنة			
المعيار	**Python**	**C#**	**(VB.NET)**
الأداء	أداؤها أقل نسبيًا من اللغتين C# و VB.net	أداء عالي نسبيا وأعلى من بايثون وفيجوال بيسك ولكنها قد تكون أقل كفاءة من بعض اللغات الاخرى وهي مناسبة للتطبيقات عالية الأداء مثل الألعاب التي تعتمد على برنامج Unity	متوسطة الأداء، حيث غالبا ما تكون البرامج النهائية أقل كفاءة وأبطأ وأكبر حجماً مقارنة باللغات منخفضة المستوى.
الأجهزة الخاصة المستخدمة			
مجالات التطبيق المفضلة			
سهولة التطوير			
وقت التطوير			
الأجهزة والبرامج اللازمة للتطوير والتشغيل			

❖ يقارن بين اللغات المختلفة من حيث البنى والتقنيات والتنفيذ المستخدمة (مع ضرب الأمثلة من نظام المكتبة) كالتالي :

وجه المقارنة مع التوضيح	**Python**	**C#**	**(VB.NET)**

Naji Kamamji + Ashraf Gane

الإعلان عن المتغيرات : تخبر هذه العملية البرنامج بنوع البيانات التي سيخزنها المتغير وتخصص له الذاكرة.	لا حاجة لإعلان صريح؛ المتغير يُنشأ بالتعيين	تعتمد على نمط C (C-Style): تحدد النوع أولاً ثم الاسم، وتستخدم الفاصلة المنقوطة : مثل : public string Title	تعتمد على الكلمات المفتاحية: تستخدم كلمات إنجليزية واضحة مثل Dim و As.مثال: Dim Title As String.
تعيين القيم : التعيين هو إعطاء قيمة لمتغير سواء كانت رقمية أو نصية أو غيرها			
الإدخال			
الإخراج			
العوامل المنطقية (إن وجدت)			
النصوص والسلاسل			
الاختيار (if,switch,…)			
التكرار (for,while,foreach…)			
الدالات / الإجراءات			
هياكل (تراكيب) البيانات (list,struct,stack,array,dictionary,..)			
معالجة الملفات			
معالجة الأحداث (إن وجدت)			

Naji Kamamji + Ashra

التوثيقات			
الفئات (الكلاس) :			
والباقي ما ترونه مناسبا			
ا			

AP3 : شرح كيفية استخدام مبادئ تصميم البرامج لإنتاج تطبيقات برمجية عالية الجودة تلبي الاحتياجات

❖ **يشرح مبادئ المنطق المطبقة على تصميم البرنامج (نظام المكتبة) :**

- **المنطق :** هو مجموعة المبادئ الموجودة في العناصر المختلفة في البرنامج والتي تسمح للبرنامج بالعمل حيث يحدد كيفية تصرف البرامج والإجراءات التي تقوم بها. وتشمل الآتي:

1- التكرار – هو منطق برمجي يتحكم في تكرار الكود ويحدد متى يجب أن ينتهي التكرار في أثناء تشغيل البرنامج .

- **التطبيق على النظام** : نلاحظ أنه تم تطبيق هذا المبدأ في أكثر من مكان في الكود مثل :

```
foreach (var b in books)
    {
      sw.WriteLine($"{b.Key}|{b.Value.Title}|{b.Value.TotalCopies}|{b.Value.AvailableCopies}");
    }
```

المنطق هنا : طالما يوجد عنصر آخر للكتب، كرر نفس العملية ،أي كرر نفس العملية (طباعة تفاصيل الكتاب) لكل عنصر يتعلق بـ Books .

2- المنطق الرياضي - ..... تعريف مختصر :

- **التطبيق على النظام :** نلاحظ هذا بوضوح في المقارنات في الجمل الشرطية (if / switch) وجمل التكرار (while) وغيرها ،مثل :

Naji Amami + Ashraf Gam

```
if (!books.ContainsKey(id))
{
   Console.WriteLine("Book not found.");
   return;
}
```

الاستدلال المنطقي هنا : إذا لم يكن الكتاب موجودًا إذا لا يمكن إعارتُه أو إرجاعه.

3- المنطق الديناميكي الافتراضي :.....الخ .

4- استخدام المجموعات: .....الخ

❖ **يشرح مبادئ التصميم والبرمجة التي تجعل البرنامج "عالي الجودة" ويلبي احتياجات المستخدم:**

مثلا : يؤثر تصميم وتنفيذ الحل البرمجي على جودته من نواحٍ مختلفة مثل:

- **الكفاءة والأداء -** وهو قياس موارد النظام التي يستهلكها البرنامج مثل الذاكرة ووقت المعالجة ...الخ .

  التطبيق على النظام : استخدام هياكل مناسبة لتسريع الأداء مثل stacks وQueue وDictionary ...

- **القابلية للصيانة -** هي سهولة تغيير الكود من قبل المبرمجين ...الخ .

  التطبيق على النظام : الكود منظم باستخدام دوال وفئات(كلاسات) وكذلك التوثيقات بمختلف أنواعها ...

- **القابلية للنقل -** .....
- **الموثوقية -** .....
- **المتانة -** .....
- **سهولة الاستخدام -** .....

❖ **يكتب المراجع في نهاية التقرير :**

مع امنياتي لكم بالتوفيق

Naji Kamanji + Ashraf Ganem